LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-410856

GT Merge Process: Version 2.0

M. P. Flanagan, D. Dodge, S. C. Myers

February 26, 2009

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

GT Merge Process: Version 2.0

Megan P. Flanagan, Doug Dodge, and Stephen C. Myers

Table of Contents

Introduction

This document summarizes the process used to merge GT25 and better data between LANL and LLNL. The merge also includes OUO arrivals provided by AFTAC for events in the merge. The merge process is mostly automated and includes extensive quality control operations at each step. Events in common between the labs are identified and resolved using GT level criteria. Arrivals in common between the labs are also resolved through the use of agreed upon arrival author rankings. Finally, baselined origin times are computed for all crustal events using either teleseismic *P*-arrivals and the *iasp91* model or, in certain regions, regional *P*-arrivals and regional velocity models that are known to be consistent with teleseismic *iasp91 P*-wave predictions.

Installation Steps

The results of the GT merge are contained in an Oracle 9 dump file named *gtmerge.dmp*. This file should be imported into an empty schema. Failure to do so will likely cause data corruption and make unusable both the merge data and any data that existed in the schema prior to the merge. The objects in the dump file were created in the table space GTMERGE_DATA and indexes were created in the table space GTMERGE_INDEX. Therefore, these table spaces must exist and have sufficient space available for the import to succeed. The GTMERGE_DATA table space must be at least 6.0 Gbytes and the GTMERGE_INDEX must be 7.5 Gbytes.

Executive Summary

We combine the core tables from each contributor and resolve unique and common GT events between contributors. Next, we merge at the pick level so that each distinct *EVENT-STATION-PHASE* tuple has a unique arrival. All BMEB (Bondar-Myers-Engdahl-Bergman) GT are re-calculated and evaluated for adherence to their criteria. Finally, new origin times are computed (baselining) for the merged GT events.

In addition to the reconciliation of events and picks between contributors, the merge process involves several quality control steps that are intended to remove outlier and irrelevant data from the final results.

The process is described in the section entitled “Merge Steps”.

Merge Results

At the completion of the merge there were a total of 92615 surviving GT events. A break-down of these numbers by GT level is shown in the tables below.

Table 1. Final GT epicenter data.

GT LEVEL	NUMBER	GT LEVEL	NUMBER
25	39014	0.7	1
5	31949	1.4	1
20	9190	2.3	1
15	9127	2.6	1
2	2056	6.1	1
0	853	7.9	1
1	315	12.7	1
3	27	12.1	1
10	24	11.8	1
0.5	10	9.6	1
0.2	8	8.2	1
0.4	6	7.6	1
0.6	5	2.9	1
0.1	4	2.7	1
4.9	3	2.4	1
0.3	2	1.5	1
3.4	2	1.2	1
1.6	2	0.8	1

Fate of Contributed GT

The two labs contributed a total of 156077 GT_EPI rows. After merging the data, 108564 unique events were identified. The number of distinct GT events was later reduced to 92615 because 15949 BMEB GT solutions did not meet their criteria after arrival cleanup and phase validation.

Before removal of bad BMEB GT there were

- 47432 events with only a LLNL contribution.
- 13786 events with only a LANL contribution.
- 47346 events with a contribution from each lab.

For 38 events, 2 of the LANL-supplied GT solutions were merged together. For 128 events, 2 of the LLNL-supplied GT were merged, and for one event, 3 LLNL-supplied GT were merged.

Relevant SQL:

Get events in Common:

```
select evid from gt_epi_ex where contrib_org = 'LLNL'
intersect
select evid from gt_epi_ex where contrib_org = 'LANL'
```

Get LLNL-only Events:

```
select evid from gt_epi_ex where contrib_org = 'LLNL'
minus
select evid from gt_epi_ex where contrib_org = 'LANL'
```

Get LANL-only Events:

```
select evid from gt_epi_ex where contrib_org = 'LANL'
minus
select evid from gt_epi_ex where contrib_org = 'LLNL'
```

Get removed GT Events:

```
select evid from gt_epi_ex
minus
select evid from gt_epi
```

Of the 108564 events, 95674 of the gt_epi_ex solutions were BMEB solutions. There was no attempt made to choose a preferred among these. Instead, after phase validation and renaming, the solutions were recalculated using all the available, relevant phase picks. Of the 12890 contributed non-BMEB solutions, 12254 were contributed by LLNL and1608 were contributed by LANL. There were 972 of these in common.

The resolution of these common events can be seen in the COMPETING_GT_RESOLUTION view. There are 976 rows in this view, because for four of the events, LLNL had two solutions that were rejected in favor of a single LANL solution. Table 2 below shows a summary of the rejected LLNL GT rows and table 3 shows a summary of the rejected LANL GT rows. Note that in a number of cases the rejected LANL rows have the same GT level as the competing LLNL rows. The reason for this is that the LLNL rows were loaded first, and if two solutions being compared had the same level and were within the GT level of one-another, then the first solution was chosen.

Table 2 Summary of rejected LLNL GT rows.

LLNL-LEVEL	LLNL-method	LANL-LEVEL	LANL-METHOD	COUNT
5	thurber_pub	1	SPOT satellite images, teleseismic epicenter est.	82
5	Engdahl_Bergman	1	known locations, map est. Degelen adit complex	77
5	Engdahl_Bergman	0	known locations, map est. Degelen adit complex	56
15	50/90	5	general explosion	18
15	50/90	5	double-difference, fault validation	14
15	50/90	5	comparisons of seismic to actual PNE locations	11
5	Engdahl_Bergman	1	SPOT satellite images, teleseismic epicenter est.	10
5	Engdahl_Bergman	1	IKONOS satellite imagery, seismogram alignment	9
15	50/90	5	Location uncer. from Sultanov et al. (1999) to GT	9
25	sultanov_pub	5	comparisons of seismic to actual PNE locations	9
5	thurber_pub	0	SPOT satellite images, teleseismic epicenter est.	6
5	Engdahl_Bergman	2	double-difference analysis of regional/tele data	5
25	sultanov_pub	5	Location uncer. from Sultanov et al. (1999) to GT	5
3	skorve_pub	1	known nuclear explosion	4
5	Engdahl_Bergman	5	teleseismic relocation, depth determination	3
1	satellite image	1	SPOT satellite images	1
5	Engdahl_Bergman	0	SPOT satellite images, teleseismic epicenter est.	1
5	InSAR	5	InSAR and seismic methods	1
15	relativeLoc	5	teleseismic relocation, depth determination	1
25	sultanov_pub	5	general explosion	1
15	50/90	5	teleseismic relocation, depth determination	1
5	Kremenet_pub_2001	3	mine collapse	1
5	Engdahl_Bergman	5	commerical satellite imagery, tele P-wave arrivals	1
1	sultanov_pub	0	SPOT satellite images, teleseismic epicenter est.	1

Table 3 Summary of rejected LANL GT rows.

LANL-LEVEL	LANL-method	LLNL-LEVEL	LLNL-METHOD	COUNT
0	DOE/other agencies, known US nuclear explosions	0	From Springer Catalog	374
15	cluster analysis	5	Engdahl_Bergman	152
1	Location uncer. from Sultanov et al. (1999) to GT	1	sultanov_pub	44
1	Soviet peaceful nuclear explosions	1	sultanov_pub	14
5	general explosion	5	Marshall_1994rel_loc	14
5	general explosion	3	skorve_pub	13
1	estimated by geodectic means or high-res maps	1	sultanov_pub	12
0	DOE/other agencies, known US nuclear explosions	0	From Springer, changed level	9
2	chemical explosions	2	constrained by mine size	3
5	comparisons of seismic to actual PNE locations	1	sultanov_pub	2
10	MSU, 5 sta 3 deg, lt 180 gap sta w/in 5 deg	5	Engdahl_Bergman	2
1	SPOT satellite images	1	satellite image	1
3	satellite imagery, personal communication	2	Oct 9, 2006 NK	1
5	general explosion	0	refraction shot	1
5	general explosion	1	gps	1
15	cluster analysis	15	50/90	1
5	teleseismic relocation, depth determination	5	Marshall_1994rel_loc	1
5	satellite shaft locations, JHD locations	1	satellite image	1
5	general explosion	2	pidcgt0	1
5	commerical satellite imagery, tele P-wave arrivals	5	Engdahl_Bergman	1
1	general explosion	1	sultanov_pub	1

- Out of the 92615 GT events, there were a total of 40723 baselined origin solutions computed. Of these, 38835 were baselined using teleseismic P and 1888 were baselined using regional arrivals. There were 23039 events that were too deep to be used and 28853 events did not have enough observations for teleseismic baselining and which could not be baselined using regional phases.

Relevant SQL:

```
select baseline_type, count(*) from best_solution group by baseline_type
```

Arrivals

A total of 21,121,311 arrivals were contributed by the two labs. In addition AFTAC supplied 488,053 arrivals that associated to the events contributed by both labs. Of these, 2,854,651 arrivals were removed for QC reasons. Many *evid-sta-phase* tuples had multiple arrivals (primarily bulletin picks contributed by both labs), so after selecting preferred arrivals for each *evid-sta-phase* there were 12,472,424 remaining at the completion of the merge. Table 4 shows the break-down by phase.

Table 4. Unique arrivals for all GT events.

IPHASE	**COUNT**
P	8341939
Pn	1537541
Pg	558174
Sn	454970
pP	406224
Sg	380433
S	288651
sP	183950
PcP	164341
Lg	154521
sS	867
PmP	583
Pdif	120
Sb	64
Pb	46

For events with a baselined origin time, there were 6,908,978 unique arrivals. The break-down by phase is shown in Table 5.

Table 5. Unique arrivals for all GT events with baselined origin times.

IPHASE	COUNT
P	5373816
Pn	710632
Sn	209935
pP	203597
sP	117354
PcP	107269
Pg	66214
S	47570
Lg	42798
Sg	28919
sS	704
Pdif	92
Sb	47
Pb	21
PmP	10

Merge Steps

The merge is made in the GTMERGE schema, and it is written in PL/SQL and Java. Each of the steps is presented in detail in this section.

Copy GT

This procedure copies data from the (pre-loaded input tables) into the appropriate GTMERGE tables and assigns the specified source name to the data. New IDs are assigned and the mapping to the source tables is created. As each event is processed, the code looks for a match with an already-loaded event. If one is found, its evid is assigned to the new data. Otherwise, a new evid is selected from the EVID sequence and assigned to the new data.

At the end of this process, all the *ids* will be new and unique to the target GTMERGE schema. Additionally, remap columns are written that allow tracing of *IDs* from the GTMERGE schema back to the original schema tables.

Note, tables with the '_EX' ending are 'extended' tables. This is simply the standard NNSA format table with extra columns at the end to allow quick remapping back to original LANL or LLNL table information. This is an alternative to building additional remap tables that might slow down processing efficiency.

Apply Arrival Cleanups

A series of procedures are called here for doing QC on the arrival data supplied by the source organization. These Are:

Remap private station codes

In a few cases, data provided by a contributor is referenced to a station code that conflicted with a different station with the same NEIC station code. In those cases, an entry was made in the SITE table with a new code for the affected data. In this step, the contributor's data using the old code are updated to use the replacement code.

Update Iphase from Phase

For cases where assoc.phase is not the same as the arrival.iphase, we copy the assoc phase value to arrival iphase.

Make Standard Phase Names

Use a phase map table to map all arrival iphase and assoc phase values to a consistent set of names.

Remove Discrepant Arrivals

Arrivals with times more than 10 s before or 3000 s after origin time are assumed to be erroneous or extraneous and are removed and written into the REMOVED_ARRIVAL table.

Remove Unwanted Phases

Any arrivals with IPHASE values not found in the ALLOWABLE_PHASE table are moved to the REMOVED_ARRIVAL table.

Set Default Deltim Values

(For arrivals with deltim unset) Here, each allowable phase arrival with *deltim* ≤ 0.0 or which is a LANL arrival with a LANL default value for deltim is set to a value of:

- 1 s for *P*-wave phases,
- 3 s for *S*-wave phases,
- 5 s for *Lg* phases.

Remove Unusable Phases

For arrivals where deltim was not set and no rule existed to set deltim, move the arrival to the REMOVED_ARRIVAL table.

Remove Distant Arrivals

Any *P*-wave phases at distances greater than 114º and any *S*-wave phases at distances greater than 19º are removed. NOTE: The code was modified after the fact to allow PKP to 180º, but no such data exists in the current results.

Update Assoc Info

Update the DELTA, SEAZ, ESAZ and possibly the AZRES field for all assoc rows.

Merge Solutions

After all data from contributing sources has been loaded into the GTMERGE schema, the MERGE_SOLUTIONS procedure is run. This procedure assigns each event to one of three categories:

RESOLVED
MANUAL_RESOLVE
RECALC_ORIGIN_RESOLVE

Non-BMEB events with only one contribution are written to the resolved table. Events in which all contributions are of type BMEB are written to the RECALC_ORIGIN_RESOLVE table. This indicates that after all arrival processing is complete, the software will bring together all arrivals relevant for the GT category and attempt a new solution. If the solution fails or if it does not meet the criterion for that category then the event will not be processed further. For cases where there are at two non-BMEB solutions with the same GT level but with positions differing by more than the GT level, a row is written into the MANUAL_RESOLVE table. As the name implies, these must be sorted out by inspection.

Managing no-site arrivals

After arrivals are loaded for a source, the procedure TABULATE_MISSING_SITES is executed. This procedure populates the MISSING_SITE table with the ARID values for arrivals that did not join to SITE. By examining the MISSING_SITE_VIEW it can be seen which SITE epochs need to be added or extended. After modifying the SITE table, the TABULATE_MISSING_SITES can be run again to verify that all arrivals can be used. If at the end of this process, there are still rows in MISSING_SITE and it is concluded that the arrivals are unimportant, we run the REMOVE_NO_SITE_ARRIVALS procedure. This procedure simply removes rows from ARRIVAL_EX found in the MISSING_SITE table and writes them to REMOVED_ARRIVAL.

Phase Validation

This Java code removes discrepant arrivals by comparing travel-time error statistics to a removal threshold. The code uses a *tau-p* formulation and the theoretical travel time and travel time uncertainty from the *iasp91* model. The statistic computed is the travel-time residual normalized by expected fluctuations in travel-time prediction and measurement error. For each *arrival* in ARRIVAL_EX, the residual is computed as the observed travel time minus the theoretical iasp91 travel. This is normalized by an error term that includes the model error, origin time uncertainty, and pick uncertainty. The origin solution from ORIGIN_EX or else information from GT_EPI_EX and GT_TIME_EX (for unassociated arrivals) and SITE information are used in determining the observed travel-time.

For each arrival, the theoretical travel time (T_{iasp}) and travel time uncertainty (E_{iasp}) from the *iasp91* model are computed. We include as an additional source of error the uncertainty in the origin time. This is the E_{origin} term which is set to a constant value of 2.0 seconds. We then take the observed travel time (T_{obs}) and pick error (*deltim*) and compute the following:

$$\frac{(T_{obs} - T_{iasp})}{\sqrt{[(E_{iasp})^2 + (deltim)^2 + (E_{origin})^2]}}$$

If this is <=3 then we keep the arrival. If not we remove the arrival and write it to the REMOVED_ARRIVAL table.

Rename Phases

This procedure is part of the ARRIVAL_CLEANER package and is run in the PL/SQL environment. The following phase names are checked, and changed if necessary, for consistency and to allow for more easily preparing KBCIT input files to compute correction surfaces for: *P, Pg, Pn, Sg, Sn, Lg*.

For all *evids* in GT_EPI_EX, the *phase* and *delta* for each *arid* in ARRIVAL_EX and EVENT_ARRIVAL_ASSOC is checked and renamed following these rules:

```
For iphase = 'P' and delta between 1.5 and 15 and depth <= 35     ===> Rename to 'Pn'
For iphase = 'P' and delta < 1.5 and depth <= 35                  ===> Rename to 'Pg'
For iphase in ( 'Pn', 'Pg') and delta > 15                        ===> Rename to 'P'
For iphase = 'S' and delta between 1.46 and 15 and depth <= 35    ===> Rename to 'Sn'
For iphase = 'S' and delta < 1.46 and depth <= 35                 ===> Rename to 'Sg'
For iphase = 'Sn' and delta > 15                                  ===> Rename to 'S'
For iphase = 'Sg' and delta > 1.46                                ===> Rename to 'Lg'
For iphase = 'Lg' and delta <= 1.46                               ===> Rename to 'Sg'
```

Merge Arrivals

This is the selection of the best arrival for every *evid-sta-phase* tuple. A unique *arid* is chosen based on author ranking in the ARRIVAL_AUTH_RANK_SINGLE table, and the *is_preferred* field in the ARRIVAL_EX table is set to '*y*'. *Arids* not chosen remain in the ARRIVAL_EX table but their *is_preferred* field is set to '*n*'. These picks are not removed to REMOVED__ARRIVAL because they have passed all other QC tests and are essentially good data.

Recalculate BMEB Solutions

This procedure drives off the RECALC_ORIGIN_RESOLVE table. For each *evid* in that table, the procedure brings together first-arriving *P* appropriate for the putative GT level and calculates a new origin solution. This solution is examined to determine whether it in fact meets the criteria. If so, a new GT_EPI_EX row is created (along with ORIGIN_EX, ASSOC, ORIGERR, and PREDICT_TT rows). The *evid* is then removed from RECALC_ORIGIN_RESOLVE and placed in the RESOLVED table. If the new solution does not meet the criteria then rows are written to FAILED_BMEB_RECALC and to FAILED_BMEB_RECALC_DETAIL and the row is removed from RECALC_ORIGIN_RESOLVE.

Teleseismic Origin Time Baselining

Here, we produce origin solutions constrained by epicenters from GT_EPI_EX and either *depth* from GT_DEPTH_EX or else by the *depth* associated with GT_EPI_EX through a linked origin in ORIGIN_EX. Baselining addresses two issues: it makes the regional travel times consistent with teleseismic times, and since the *Pn* tomography code does not include relocating the events when inverting for the *Pn* velocities this step insures consistent origin times with which to measure *Pn* residuals.

All available teleseismic ($\Delta \geq 22^\circ$) *P* phases are used in the solution. If the event does not have 10 or more teleseismic *P* picks then it is not used. Only shallow events are baselined. The depth cutoff used is 35 km. Several thousand events outside of regions with known deep seismicity had depths greater than 35 km. Since these depths are probably erroneous, we fix the depth to 10 km for those events before baselining them.

REGIONAL ORIGINTIME BASELINER

This java program produces baselined origin solutions with origin times determined from travel time residuals calculated using specific regional models. We used this code to process a subset of the GT that could not be baselined using the teleseismic baseliner because there were not enough teleseismic P observations.

Like the teleseismic baselining code, the regional baseliner fixes the origin solutions to have the same epicenter and depth as the GT solution being baselined. It then computes the origin time as a weighted average of the difference between the arrival times and predicted travel times using an appropriate regional model. The new origin time *O* is given as:

$$O = \sum_{i}^{N} \omega_i (a_i - T_i)$$

Where

$$\omega_i = \frac{1}{\sigma_i \sum_{j}^{N} \frac{1}{\sigma_j}} \text{ and } \sigma_i = \sqrt{deltim_i^2 + ModelError_i^2}$$

We used the phases Pn, Pg, and P and required a minimum of three such observations to baseline an event. For European Arctic events we use the EA model. For the China region we used the LMV model, and for the YSKP region we used the RBHmod model.

Appendix 1: GT Merge Schema

The GTMERGE schema and ER diagram can be found in the attached **gtmerge.DataModel** folder which contains several *.html* files which can be opened in a web browser. Open the **index.htm** file first and all subsequent links can be found in this master file.

Note, this may not work with the Safari browser, but has been tested successfully with the Internet Explorer, Firefox, and Mozilla browsers.

Descriptions of the tables and views used in the merge are found in Tables 7 and 8 below. Table 7 lists tables and views that are used by codes during the merge process, but that are of limited usefulness to end-users of the merge. Table 8 lists tables and views that are expected to be useful in using and interpreting the merge results.

Table 7. Tables and Views Used by the Merge Codes

TABLE_NAME	COMMENTS
ALLOWABLE_PHASE	This is a table that contains the list of phase names for arrivals that are to be retained in the merge process. In the case of LLNL arrivals, filtering occurs on the way into the merge so that arrivals that are not in allowable phase do not exist anywhere in the merge schema. In the case of LANL arrivals, the arrivals are loaded without filtering. The phase names may then be remapped if not in allowable phase. e.g. (P) --> P. After remapping, any arrivals with phase names not in allowable phase are moved to REMOVED_ARRIVAL.
ARRIVAL_AUTH_RANK_SINGLE	This table contains the arrival author ranks used in selecting a preferred arrival among multiple contributions for a single evid,sta,iphase
ARRIVAL_EX	ARRIVAL_EX contains all the columns of the CSS3.0 ARRIVAL table as well as additional columns relevant to the merge process. The columns were added to the arrival table (instead of being used to form a lookup table) in order to improve the performance of several merge procedures.
BEST_EVENT_INFO_TMP	This is an Oracle global temporary table used during the phase validation process to temporarily store the best estimates for all event parameters for the purpose of predicting arrival times. It will contain no rows when the merge is complete.
DEEP_FOCUS_REGION	This table (and the associated DEEP_FOCUS_REGION_DATA table) defines a number of regions where deep seismicity is known to occur. During teleseismic baselining, events with hypocenter depths greater than 35 km are checked to see if the epicenters are within one of these regions. If so, the event is rejected for baselining. Otherwise, the hypocenter is fixed at 10 km and processing continues.
DEEP_FOCUS_REGION_DATA	This table (and its parent DEEP_FOCUS_REGION table) defines a number of regions where deep seismicity is known to occur. During teleseismic baselining events with hypocenter depths greater than 35 km are checked to see if the epicenters are within one of these regions. If so, the event is rejected for baselining. Otherwise, the hypocenter is fixed at 10 km and processing continues.
EVENTS_TO_CALIBRATE	The EVENTS_TO_CALIBRATE table contains the EVIDs of the unique events identified during the merge process. It also serves to enforce referential integrity among the tables that have EVID as a column.
EVENT_ARRIVAL_ASSOC	Associates arrivals with events regardless of whether the arrival has been used in a specific solution.
GT_DEPTH_EX	This table is the NNSA GT_DEPTH table expanded by two columns used during the merge and also useful for drilling down in the results. The additional columns are ORIG_DEPTHID and ORIG_EVID. These columns along with CONTRIB_ORG uniquely identify the relevant columns in the originator schema(s).
GT_EPI_EX	The GT_EPI_EX table is the NNSA GT_EPI table augmented with two columns useful in producing the merge and in drilling down into the results. GT_EPI is available as a view into this table.
GT_METHOD_RANK	This table is used only for selecting the best GT from the LLNL.GT_EPI table. For a given level of gt there may be more than one estimate and this table is used to discriminate among the candidates.
GT_TIME_EX	The GT_TIME_EX table is the NNSA GT_TIME table with two additional columns added that are useful in producing the merged data set and in drilling down into the results. GT_TIME is available as a view.
INPUT_ARRIVAL	This is a CSS arrival table used in loading arrivals into the GTMERGE schema. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables. Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.

INPUT_ASSOC	This is a CSS ASSOC table used to hold contributed ASSOC rows for loading into the GTMERGE schema. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.
INPUT_EVENT_ARRIVAL_ASSOC	This is a LLNL EVENT_ARRIVAL_ASSOC table used to hold event to arrival associations for data being loaded into the GTMERGE schema. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables. Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.
INPUT_GT_DEPTH	This is a NNSA GT_DEPTH table used to hole contributed GT_DEPTH rows for loading into the GTMERGE schema. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables. Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.
INPUT_GT_EPI	This is a NNSA GT_EPI table used to hold contributed GT_EPI solutions for loading into the GTMERGE schema. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables. Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.
INPUT_GT_TIME	This is a NNSA GT_TIME table used to hold contributed GT_TIME rows for loading into the GTMERGE schema. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables. Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.
INPUT_ORIGERR	This is a CSS ORIGERR table used to hold contributed ORIGERR data to be loaded into the GTMERGE schema. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables. Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.
INPUT_ORIGIN	This is a CSS ORIGIN table used to input contributed origin solutions. Because the merge code is written in PL/SQL, the tables it references must exist at compilation time. Rather than recompiling the code for each contributor's data, the code is written to select input from these tables. Prior to running the merge code, an input data set must first be loaded into the input_xxx tables.
MANUAL_BASELINE	The MANUAL_BASELINE table is used to drive the regional baselining process. After teleseismic baselining is complete the procedure JAVA_GT.POPULATE_MANUAL_BASELINE is run. This selects all the unbaselined events that could not be baselined because of a lack of teleseismic P and that lie within one of the regions in the REGION_POLYGON table. These are added to the MANUAL_BASELINE table. Not all events in MANUAL_BASELINE will end up being processed because there may not be enough regional picks to do baselining either. This is not intended as an end-user table.
MANUAL_RESOLVE	This table contains the EVIDs of events that had GT from both labs and that could not be resolved automatically. For these events, both labs have a solution with the same GT level and at least one of the solutions is a non BMEB type. The MANUAL_RESOLVE_VIEW contains a useful side-by-side comparison of the competing GT. At the end of the merge, this table is expected to be empty.

MISSING_SITE	The MISSING_SITE table contains the ARID values for arrivals for which no SITE information was found. It is populated after each contributor's arrivals are loaded with the expectation that the missing entries will be provided. During arrival clean up, any arrivals found in this table are moved into the REMOVED_ARRIVAL table and are not processed further. This is not intended for end-users.
ORIGIN_EX	The ORIGIN_EX is the CSS ORIGIN table augmented with columns linking origin keys to their original values in their source schemas. Also, the IS_PREFERRED column identifies the solution corresponding to the best GT for the event, or when the event has been baselined, the baseline origin row. This is not intended as an end-user table.
RECALC_ORIGIN_RESOLVE	The RECALC_ORIGIN_RESOLVE table contains the EVIDs of events for which both labs provided BMEB GT and which consequently need to be recalculated. Prior to baselining, for each EVID in this table, all relevant arrivals are gathered and a new solution calculated and compared against the criterion. If the new solution passes, the new solution is written and made the preferred solution. Otherwise, entries are made in the FAILED_BMEB_RECALC table. In either case, the EVID is removed from this table. At the end of the merge, there should be no rows in this table. This is not intended as an end-user table.
REGION_POLYGON	This table defines the polygons for regions in which regional baselining is to be attempted. After teleseismic baselining is complete, the unbaselined events that did not have enough teleseismic arrivals and that fall within one of the regions in this table are selected for regional baselining. The model to use for each event is chosen based on the polygon containing the epicenter.
RESOLVED	The RESOLVED table contains the EVIDs and EPICENTERIDs of all the events that have unique and valid GT. The merge code uses this table to track which events have had thier GT validated and in the case where there is more than one contribution, had a single contribution selected. Only events listed in this table are selected for baselining. It is not intended for end users of the data.
SITE	This is a CSS SITE table supporting all arrivals used in the merge. The table is a merge of information from multiple sources including neic, ISC, LLNL, LANL, IRIS, AFTAC. Currently a limitation of this table is that the array-specific columns are not populated.
UNBASELINED_GT	This is a table used by the merge code that holds the EVIDs and reasons for GT that could not be baselined. The table is populated by the teleseismic origin time baseliner. It is then used to populate the MANUAL_BASELINE table for regional baselining. It is also used in constructing the BEST_SOLUTION view.
MANUAL_RESOLVE_VIEW	The MANUAL_RESOLVE_VIEW is used during the GT merge processing to help identify which of two conflicting GT_EPI rows should be retained in the merge. It is of no use to users of the merge, because when the merge is complete, there are no rows displayed in this view.
MISSING_SITE_VIEW	The MISSING_SITE_VIEW is used during the merge process to identify SITE rows that are missing or that need epochs added or extended. By showing the count of unusable arrivals for each sta, it can help in prioritizing work on the SITE table. This view is of no use to end users since it is empty when the merge is complete.

Table 8. Tables and Views for End-users of the Merge

TABLE_NAME	COMMENTS
ASSOC	The assoc table associates arrivals to the original contributed origin solutions, to the re-calculated BMEB solutions, and to the baselined origin solutions.
CAL_ORIGIN	The CAL_ORIGIN table contains references to all the base-lined origin solutions. In addition to identifying all such origins, it contains references to the data used to constrain the solutions.
EVENT_ARRIVAL_ASSOC	Associates arrivals with events regardless of whether the arrival has been used in a specific solution.
FAILED_BMEB_RECALC	The FAILED_BMEB_RECALC table contains information on attempted recalculation of BMEB GT that failed.
FAILED_BMEB_RECALC_DETAIL	This table contains a subset of the columns of ASSOC for attempted BMEB solutions that were not written because they failed the test for the specific GT level. Given an entry from the FAILED_BMEB_RECALC table, the row subset for the failed event can be manually inspected here.
FAILED_VALIDATION	The FAILED_VALIDATION table contains details of inputs to and outputs from the travel time calculator for arrivals that failed phase validation.
ORIGERR	This is a CSS ORIGERR table. It contains entries from contributed solutions as well as from BMEB solutions and baseline solutions computed in the merge.
PREDICT_TT	This table provides additional information about the predictions for each arrival processed by the LOCOO JNI locator. All arrivals used in BMEB locations and/or in teleseismic baselining will have an entry in this table.
REMOVED_ARRIVAL	The REMOVED_ARRIVAL table contains entries from the ARRIVAL_EX table that were removed during processing. Each removed arrival also has a reason for its removal.
REMOVED_ASSOC	The REMOVED_ASSOC table contains rows originally in the ASSOC table for arrivals that have been removed for QC purposes. This table was added based on feedback from the AFTAC merge, but after the OUO merge had run and so was populated after the fact. There may be some missing rows for data loaded from flat files.
REMOVED_EVENT_ARRIVAL_ASSOC	Associates removed arrivals to the event with which they were originally associated. This table was added based on feedback from AFTAC and after the OUO merge was substantially complete. It was populated after the fact, and some associations may not have been retrieved.
SITE	This is a CSS SITE table supporting all arrivals used in the merge. The table is a merge of information from multiple sources including neic, ISC, LLNL, LANL, IRIS, AFTAC. Currently a limitation of this table is that the array-specific columns are not populated.
ARRIVAL	The ARRIVAL view contains just the CSS 3.0 ARRIVAL columns and only has rows for the preferred arrivals in ARRIVAL_EX
ARRIVAL_REMAP	The ARRIVAL_REMAP view shows the link between an arrival in the GTMERGE schema and the original arrival in the contributor schema.
BEST_SOLUTION	The BEST_SOLUTION view shows the best available origin information for each event in the merge with surviving GT solutions. This view contains both baselined and non-baselined solutions, and identifies which are baselined and which are not, the baseline types (for baselined rows) and the reason for not baselining for the non-baselined rows. It shows the relevant GT information for (epicenter, depth, time) as available.
COMPETING_GT_RESOLUTION	The COMPETING_GT_RESOLUTION view shows a comparison of competing non-BMEB GT data by EVID. Each event for which more than one non-BMEB GT_EPI row was contributed shows the (source, GT-level, method) of the preferred GT and of the rejected GT.
EVENT	The EVENT view simulates the CSS 3.0 EVENT table. It has one row for each event with a surviving GT_EPI row whether baselined or not. The

	PREFOR is the ORID of the preferred origin from ORIGIN_EX.
FAILED_PHASE_VALIDATION_VIEW	The FAILED_PHASE_VALIDATION_VIEW contains all the information used by the phase validation software to reject a particular arrival. This view was created mainly to observe the behavior of the phase validation software, but may have some utility to users of the GT merge.
GT_DEPTH	The GT_DEPTH view contains only the columns from the NNSA GT_DEPTH table and only the preferred rows from GT_DEPTH_EX table. It is intended to serve queries that expect the column layout of the NNSA table.
GT_EPI	The GT_EPI view contains only the columns from the NNSA GT_EPI table and only the preferred rows from GT_EPI_EX table. It is intended to serve queries that expect the column layout of the NNSA table.
GT_EPI_REMAP	The GT_EPI_REMAP view shows the relation of a row in the GT_EPI_EX table to the GT_EPI row from the contributor.
GT_TIME	The GT_TIME view contains only the columns from the NNSA GT_TIME table and only the preferred rows from GT_TIME_EX table. It is intended to serve queries that expect the column layout of the NNSA table.
INVALID_GT_EPI	The INVALID_GT_EPI view contains rows from the GT_EPI_EX table for events in which the contributed GT was of type BMEB but which did not meet the criterion when the relevant data were used to re-compute a new solution.
NON_PREFERRED_ARRIVAL	The NON_PREFERRED_ARRIVAL view shows all the arrivals that passed QC tests, but that were not selected as the preferred for an EVID,STA,IPHASE. If you are looking for an arrival that did not make it through to the end of processing, this is one place to look.
NON_PREFERRED_GT_EPI	The NON_PREFERRED_GT_EPI view lists all the GT_EPI_EX rows that are not preferred for events where there is a preferred GT_EPI. This view does not show rows for those events where all the GT were of type BMEB but for which the relevant criteria were not met. Those rows can be found in the INVALID_GT_EPI view.
ORIGIN	The ORIGIN view contains the columns of the CSS 3.0 ORIGIN table selected from the ORIGIN_EX table. It only contains the preferred rows from ORIGIN_EX.
ORIGIN_REMAP	The ORIGIN_REMAP view maps the EVID and ORID from the ORIGIN_EX table to the source ORIGIN table EVID and ORID.
TRAVELTIME_VIEW	The TRAVELTIME_VIEW view is an example view that presents observed travel time for a specific phase observed at a specific station for a baselined origin solution. The view columns also contain the auxiliary data that would be necessary to calculate a predicted time.